ECEN 631 – Robotic Vision

Visual Inspection Project Guide

February 7, 2025

Project Scope:

This project is one of the three focuses of this robotic vision course. It is designed to introduce machine vision application to students through a simple real-time visual inspection project. Machine vision technology has been used for many industrial factory automation applications for the past 30+ years. Through this project, students practice implementing real-time vision algorithms and using machine vision technology for visual inspection applications.

Project Requirements:

- A conveyor belt with a speed controller is setup in 408 EB for this project.
- A Logictech webcam and Flea-2 camera are mounted on the conveyor.
- The camera field of view is approximately 6”×4.5”.
- The team selects product(s) and prepare product samples for training and inspection.
- Product samples should be smaller than 3”×3”.
- Samples should include good, acceptable (ugly), and defective (bad) products.
- The team defines its own inspection criteria.
- The system must display “Good”, “Bad”, or “Ugly” on the screen to demonstrate the performance of the system.
- There are three possible approaches for this project, 1) traditional computer vision techniques with pre-determined grading criteria, 2) hand-crafted features with machine learning techniques, and 3) deep learning.
- You will be required to use the machine learning approach.
- Grading is based on the performance and complexity of the inspection.

System:

- A Linux machine with OpenCV, PyTorch, and camera driver installed has been set up for this project in 408 EB. You should have been granted after-hour access to the room.
- Use your CAEDM login to use the computer.
- Your code and dataset will be stored in your CAEDM driver.
- Please do not install or uninstall any software or library packages locally.
- Run the Example capture file to see how to grab frames from the camera.

  Type: python3 VisualInspection.py

Software:

You can download the following sample programs from Learning Suite.

- VisualInspection: Program for capturing live image from the camera.
- LBPSVM: Sample code for calculating LBP and using SVM for classification.
- PyTorch: Sample code for using PyTorch for classification.
- TFKeras: Sample code for using TensorFlow Keras for classification (not tested).

Camera Adjustment Commands: Use these commands to adjust the settings of the USB camera.

```
#!/usr/bin/bash
v4l2-ctl -d /dev/video2 --set-ctrl=auto_exposure=1
v4l2-ctl -d /dev/video2 --set-ctrl=exposure_dynamic_framerate=0
v4l2-ctl -d /dev/video2 --set-ctrl=exposure_time_absolute=250
v4l2-ctl -d /dev/video2 --set-ctrl=white_balance_automatic=0
v4l2-ctl -d /dev/video2 --set-ctrl=white_balance_temperature=4600
v4l2-ctl -d /dev/video2 --set-ctrl=focus_automatic_continuous=0
v4l2-ctl -d /dev/video2 --set-ctrl=gain=100
```